

Lista de exercícios práticos

Instruções para o Aluno

Seu trabalho será dividido em três arquivos:

1. **funcoes.h**: Este é o arquivo de cabeçalho (header). Ele contém as declarações (protótipos) de todas as funções que você precisa implementar. Este arquivo já está pronto para você.
2. **funcoes.c**: Este é o arquivo de implementação. Você deve criar este arquivo, incluir **funcoes.h** nele e escrever o corpo (a lógica) de cada uma das funções declaradas no cabeçalho.
3. **main.c**: Este é o seu arquivo principal para testes. Você deve criar este arquivo, incluir **funcoes.h** e escrever a função **main**, que deverá chamar e testar cada uma das funções que você implementou em **funcoes.c**.

Exercício 1: Passagem por Valor – Conversão de Medidas Escreva uma função que receba por parâmetro um valor inteiro representando uma distância em metros e a converta para centímetros. A função deve retornar o valor calculado.

- **Protótipo:** `int converteParaCentimetros(int metros);`

Exercício 2: Lógica com Retorno – Verificação de Maioridade Crie uma função que receba a idade de uma pessoa como um número inteiro. A função deve retornar **1** se a pessoa for maior de idade (idade \geq 18) e **0** caso contrário.

- **Protótipo:** `int ehMaiorDeIdade(int idade);`

Exercício 3: Escopo de Variáveis – Contador de Chamadas Escreva uma função que não receba nenhum parâmetro e não retorne valor. A função deve incrementar uma variável global cada vez que for chamada. Crie um programa que chame esta função 3 vezes e imprima o valor final da variável.

- **Protótipo:** `void contaChamada(void);`

Exercício 4: Passagem por Referência – Operações Múltiplas Crie uma função que receba dois números inteiros por referência. A função deve alterar o valor da primeira variável para o seu dobro e o da segunda para o seu triplo.

- **Protótipo:** `void calculaDobroTriplo(int *a, int *b);`



Exercício 5: Passagem de Vetores – Análise de Notas Escreva uma função que receba como parâmetros um vetor de números reais (notas de um aluno) e o número de elementos no vetor. A função deve calcular e retornar a média das notas.

- **Protótipo:** `float calculaMedia(float notas[], int n);`

Exercício 6: Passagem de Matrizes – Preenchimento de Matriz Crie uma função que receba uma matriz de inteiros 3x3 e a preencha com o produto do seu índice de linha pelo índice de coluna.

- **Protótipo:** `void preencheMatriz(int matriz[3][3]);`

Exercício 7: Manipulação de Caracteres – Vogal ou Consoante Crie uma função que receba um caractere como parâmetro e retorne `1` se o caractere for uma vogal (a, e, i, o, u, maiúsculas ou minúsculas) e `0` caso contrário.

- **Protótipo:** `int ehVogal(char letra);`

Exercício 8: Vetores e Passagem por Referência – Mínimo e Máximo Escreva uma função que receba um vetor de inteiros e seu tamanho. A função deve encontrar o menor e o maior elemento do vetor e retornar esses dois valores para a função `main` usando passagem por referência.

- **Protótipo:** `void encontraMinMax(int vetor[], int n, int *min, int *max);`

Exercício 9: Lógica com Números – Soma dos Dígitos Elabore uma função que receba um número inteiro positivo como parâmetro e retorne a soma de seus dígitos. Por exemplo, se o número for `123`, a função deve retornar `6` (`1 + 2 + 3`).

- **Protótipo:** `int somaDigitos(int numero);`

Exercício 10: Modularização – Análise de Matriz Quadrada Crie um programa modularizado para analisar uma matriz quadrada 3x3 de inteiros. Para isso, crie as seguintes funções:

1. Uma função para ler os valores da matriz do usuário.
 2. Uma função para calcular a soma dos elementos da diagonal principal da matriz.
 3. Uma função para exibir a matriz na tela.
- **Protótipos:**
`void leMatriz3x3(int mat[3][3]); int`
`somaDiagonalPrincipal3x3(int mat[3][3]); void`
`exibeMatriz3x3(int mat[3][3]);`
-



Arquivo de Cabeçalho (funcoes.h)

Este arquivo contém todos os protótipos. Copie e salve este conteúdo como `funcoes.h`.

```
#ifndef FUNCOES_H
#define FUNCOES_H

// Exercício 1
int converteParaCentimetros(int metros);

// Exercício 2
int ehMaiorDeIdade(int idade);

// Exercício 3
// Declara que a variável global 'contador' existe e será definida
em outro arquivo.
extern int contador;
void contaChamada(void);

// Exercício 4
void calculaDobroTriplo(int *a, int *b);

// Exercício 5
float calculaMedia(float notas[], int n);

// Exercício 6
void preencheMatriz(int matriz[3][3]);

// Exercício 7
int ehVogal(char letra);

// Exercício 8
void encontraMinMax(int vetor[], int n, int *min, int *max);

// Exercício 9
int somaDigitos(int numero);

// Exercício 10
void leMatriz3x3(int mat[3][3]);
int somaDiagonalPrincipal3x3(int mat[3][3]);
void exibeMatriz3x3(int mat[3][3]);

#endif // FUNCOES_H
```